CLAIMS

What is claimed is:

1. A computer system providing Internet protocol security without secure domain name resolution, the system comprising:

a local domain name service (DNS) server that is communicatively coupled to a processor and that includes a secure Internet security protocol (IPSEC) cache, wherein the secure IPSEC cache is readable only by an Internet protocol (IP) processing layer of an operating system that controls execution of an application program by the processor;

a security policy data store that is communicatively coupled to the IP processing layer;

a computer-readable medium accessible to the processor and comprising one or more sequences of instructions which, when executed by the processor, cause the processor to carry out the steps of:

receiving a message generated as a result of execution of the application program and that contains a domain name;

searching the secure IPSEC cache for an entry that matches the domain name;

querying the security policy data store for an IPSEC policy matching the domain name;

applying the IPSEC policy to the message; and

purging the matching entry from the cache.

2. A computer system as recited in Claim 1, wherein the secure IPSEC cache comprises a plurality of cache entries, wherein each cache entry comprises a DNS name, one or more corresponding IP addresses, and information that uniquely associates the cache entry with a particular application process or execution time.

3. A computer system as recited in Claim 2, wherein the step of searching the secure IPSEC cache further comprises the step of searching the secure IPSEC cache for an entry that matches a process identifier of the application program, based on the information that uniquely associates the cache entry with a particular application process or execution time.

4. A computer system as recited in Claim 2, wherein the information that uniquely associates the cache entry with a particular application process or execution time comprises a process identifier value and a transaction identifier value.

5. A computer system as recited in Claim 4, wherein the step of searching the secure IPSEC cache further comprises the step of searching the secure IPSEC cache for an entry that matches a process and transaction associated with the application program, based on the process identifier value and transaction identifier value in the cache.

6. A computer system as recited in Claim 1, further comprising the step of querying the security policy database for an IPSEC policy based on an IP address that is resolved from the domain name received from the application program only when a matching cache entry is not found by searching the cache based on the domain name.

7. A computer system as recited in Claim 1, further comprising the steps of:
receiving a request to resolve a DNS name into network addresses;
resolving the DNS name using the local DNS server, resulting in generating one or more network addresses corresponding to the DNS name;
determining identifier information that uniquely associates the request with a particular application process or execution time; and
storing the DNS name, the network addresses, and the identifier information as an entry in the secure IPSEC cache.

8. A method for providing Internet protocol security without secure domain name resolution, the method comprising the computer-implemented steps of:

receiving a message generated as a result of execution of an application program and that contains a domain name;

searching a secure Internet security protocol (IPSEC) cache for an entry that matches the domain name, wherein the secure IPSEC cache is communicatively coupled to a local domain name service (DNS) server, and wherein the secure IPSEC cache is readable only by an Internet protocol (IP) processing layer of an operating system that controls execution of the application program;;

querying a security policy data store that is communicatively coupled to the IP processing layer for an IPSEC policy matching the domain name;

applying the IPSEC policy to the message; and

purging the matching entry from the cache.

9. A method as recited in Claim 8, wherein the secure IPSEC cache comprises a plurality of cache entries, wherein each cache entry comprises a DNS name, one or more corresponding IP addresses, and information that uniquely associates the cache entry with a particular application process or execution time.

10. A method as recited in Claim 9, wherein the step of searching the secure IPSEC cache further comprises the step of searching the secure IPSEC cache for an entry that matches a process identifier of the application program, based on the information that uniquely associates the cache entry with a particular application process or execution time.

11. A method as recited in Claim 9, wherein the information that uniquely associates the cache entry with a particular application process or execution time comprises a process identifier value and a transaction identifier value.

12. A method as recited in Claim 11, wherein the step of searching the secure IPSEC cache further comprises the step of searching the secure IPSEC cache for an entry that matches a process and transaction associated with the application program, based on the process identifier value and transaction identifier value in the cache.

13. A method as recited in Claim 8, further comprising the step of querying the security policy database for an IPSEC policy based on an IP address that is resolved from the domain name received from the application program only when a matching cache entry is not found by searching the cache based on the domain name.

14. A method as recited in Claim 8, further comprising the steps of:
receiving a request to resolve a DNS name into network addresses;
resolving the DNS name using the local DNS server, resulting in generating one or more network addresses corresponding to the DNS name;
determining identifier information that uniquely associates the request with a particular application process or execution time; and
storing the DNS name, the network addresses, and the identifier information as an entry in the secure IPSEC cache.

15. A computer-readable medium carrying one or more sequences of instructions for providing Internet protocol security without secure domain name resolution, which instructions, when executed by one or more processors, cause the one or more processors to carry out the steps of:
receiving a message generated as a result of execution of an application program and that contains a domain name;

searching a secure Internet security protocol (IPSEC) cache for an entry that matches the domain name, wherein the secure IPSEC cache is communicatively coupled to a local domain name service (DNS) server, and wherein the secure IPSEC cache is readable only by an Internet protocol (IP) processing layer of an operating system that controls execution of the application program;;

querying a security policy data store that is communicatively coupled to the IP processing layer for an IPSEC policy matching the domain name;

applying the IPSEC policy to the message; and

purging the matching entry from the cache.

16. A computer-readable medium as recited in Claim 15, wherein the secure IPSEC cache comprises a plurality of cache entries, wherein each cache entry comprises a DNS name, one or more corresponding IP addresses, and information that uniquely associates the cache entry with a particular application process or execution time.

17. A computer-readable medium as recited in Claim 16, wherein the step of searching the secure IPSEC cache further comprises the step of searching the secure IPSEC cache for an entry that matches a process identifier of the application program, based on the information that uniquely associates the cache entry with a particular application process or execution time.

18. A computer-readable medium as recited in Claim 17, wherein the information that uniquely associates the cache entry with a particular application process or execution time comprises a process identifier value and a transaction identifier value.

19. A computer-readable medium as recited in Claim 18, wherein the step of searching the secure IPSEC cache further comprises the step of searching the secure IPSEC cache for an entry that matches a process and transaction associated with the application program, based on the process identifier value and transaction identifier value in the cache.

20. A computer-readable medium as recited in Claim 15, further comprising the step of querying the security policy database for an IPSEC policy based on an IP address that is resolved from the domain name received from the application program only when a matching cache entry is not found by searching the cache based on the domain name.

21. A computer-readable medium as recited in Claim 15, further comprising the steps of:
receiving a request to resolve a DNS name into network addresses;
resolving the DNS name using the local DNS server, resulting in generating one or more network addresses corresponding to the DNS name;
determining identifier information that uniquely associates the request with a particular application process or execution time; and
storing the DNS name, the network addresses, and the identifier information as an entry in the secure IPSEC cache.

22. An apparatus for providing Internet protocol security without secure domain name resolution, comprising:
means for receiving a message generated as a result of execution of an application program and that contains a domain name;
means for searching a secure Internet security protocol (IPSEC) cache for an entry that matches the domain name, wherein the secure IPSEC cache is communicatively coupled to a local domain name service (DNS) server, and wherein the secure IPSEC cache is readable only by an Internet protocol (IP) processing layer of an operating system that controls execution of the application program;;
means for querying a security policy data store that is communicatively coupled to the IP processing layer for an IPSEC policy matching the domain name;
means for applying the IPSEC policy to the message; and
means for purging the matching entry from the cache.

23. An apparatus for providing Internet protocol security, without secure domain name resolution, for messages that are carried by a packet-switched data network, comprising:

a network interface that is coupled to the data network for receiving one or more packet flows therefrom;

a processor;

one or more stored sequences of instructions which, when executed by the processor, cause the processor to carry out the steps of:

receiving a message generated as a result of execution of an application program and that contains a domain name;

searching a secure Internet security protocol (IPSEC) cache for an entry that matches the domain name, wherein the secure IPSEC cache is communicatively coupled to a local domain name service (DNS) server, and wherein the secure IPSEC cache is readable only by an Internet protocol (IP) processing layer of an operating system that controls execution of the application program;;

querying a security policy data store that is communicatively coupled to the IP processing layer for an IPSEC policy matching the domain name;

applying the IPSEC policy to the message; and

purging the matching entry from the cache.